MoE アーキテクチャによる破滅的忘却の抑制効果の評価

杉本 陽大 [1]　李 宰成 [1]　吉田 倖 [1]　鈴木 潤 [1,2,3]
[1] 東北大学　[2] 理化学研究所　[3] 国立情報学研究所 LLMC
is-failab-research@grp.tohoku.ac.jp

概要

近年の大規模言語モデル（LLM）では，計算効率の向上を目的として Sparsely-Gated Mixture-of-Experts（MoE）アーキテクチャが広く採用されている．本研究では，MoE モデルが標準的な Dense モデルと比較して，事前学習時の破滅的忘却を抑制する効果がある可能性を定量的に示す．具体的には，言語モデルが事前学習時にエンティティに関する事実を学んでいると仮定して，その周辺情報の再現性を指標として忘却量を測定した．その結果，MoE のような FFN 層の重みの一部のみを利用する構造的特性が，既存知識への干渉を低減し，事前学習時の効率化に寄与する可能性を示す．

1　はじめに

破滅的忘却（Catastrophic Forgetting）[1] は，ニューラルネットワークにおける根源的な課題であり，大規模言語モデル（LLM）に限らず多くの学習モデルにおいて回避すべき問題として位置づけられてきた．特に LLM の文脈においては，教師ありファインチューニング（SFT）や強化学習（RL）といった，複数のデータセットを用いた逐次的な学習プロセスにおいてその影響が広く議論されている [2]．しかし，破滅的忘却はこうした段階的な学習に限らず，単一のデータセットを用いた学習過程においても生じ得るものであり，学習全体を通じてモデルの知識獲得の効率を低減する要因となっている [3]．

現在，LLM の開発においては，事前学習に投入するデータ量を増大させることが性能向上の鍵とされている [4, 5]．しかし，インターネット上の高品質なデータには限界が見え始めており [6]，今後は限られたデータ量から効率よく知識を記憶・定着させる「学習の質」が，性能向上の観点から極めて重要になると想定されている．

こうした背景のもと，計算コストを抑えつつモデル容量を拡大する手法として，Sparsely-Gated Mixture-of-Experts（MoE）アーキテクチャ [7, 8, 9] を採用する LLM が増加している [10, 11, 12, 13]．MoE の主な採用理由は，推論・学習時の計算量を維持したままパラメータ数を増やし，知識容量を拡大できる点にある．本研究では，図 1 に示したような，MoE の「トークンごとに特定の Expert（FFN）のみを活性化させる」という構造的特性が，既存の知識への干渉を最小限に抑え，結果として破滅的忘却を抑制するのではないかという仮説を立てた．

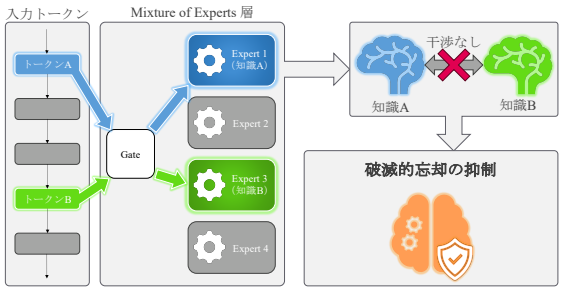

図 1　MoE による忘却抑制効果

評価の結果，MoE モデルは Dense モデルと比較して学習過程における破滅的忘却を抑制できる可能性があることが示された．この成果は，パラメータをスパースに分散させ，更新範囲を限定するアーキテクチャが，単なる計算効率の向上に留まらず，知識の保持と定着において極めて効率的な学習を可能にすることを示唆している．

2　関連研究

2.1　破滅的忘却

ニューラルネットワークが新たなタスクを学習する際，既習知識を急速に失う現象は破滅的忘却（Catastrophic Forgetting）[1] として知られている．これは，新タスクへの最適化過程で，旧タスクに寄与

していたパラメータが無造作に上書きされることに起因する．特に LLM においては，膨大なパラメータの中から特定のタスクに関わる領域を特定することは困難であり，継続事前学習や SFT，RL の各フェーズにおいて，以前の能力が損なわれることが大きな課題となっている [2].

近年，この忘却を抑制する手法が盛んに研究されている．しかし，その多くはタスク間の継続学習に焦点を当てたものである [2]．一方で，事前学習という単一のプロセス内においても，膨大なデータを逐次処理する過程で初期の学習内容を忘却している可能性は高い [3]．効率的なモデル構築を実現する上で，この事前学習フェーズにおける忘却現象の解明と抑制は不可欠と言える．しかし，既存の研究成果から得られた知見では未だ明確な解明には至っていないと言える．

2.2 MoE アーキテクチャによる忘却抑制

Mixture-of-Experts（MoE）は，パラメータ数（知識容量）を大幅に拡大しつつも推論コストを抑制できるため，近年の大規模言語モデル（LLM）において広く採用されている [10, 11, 12, 13]．理論面では，MoE の疎な活性化構造が非言語タスクにおける破滅的忘却の抑制に寄与することが知られており [14]，これを応用した新たな学習手法やアーキテクチャも数多く提案されている [15, 16, 17].

しかし，現在普及している標準的な MoE アーキテクチャが，実際の言語モデルにおいて忘却抑制にどの程度寄与しているかを実験的に示した研究は限定的である．この背景には，言語タスクにおける忘却を多角的に評価するための指標や手法が未だ十分に確立されていないという課題があると考えられる．したがって，既存の MoE モデルの忘却特性を詳細に検証することは，より堅牢な学習手法を確立する上で極めて重要な工程といえる．

2.3 事前学習時の破滅的忘却の評価

事前学習における言語モデルの評価には一般に perplexity（PPL）が用いられているが，破滅的忘却の評価としては不十分である点が Liao らの先行研究 [3] で指摘されている．PPL は文法的なパターンや頻出トークンを過大評価する傾向があり，モデルが獲得した具体的な事実的知識の欠落を感度良く捉えられないためである．この問題点を考慮して提案された指標が M_{in} および M_{ex} と言った，エンティティ周辺の知識に関する評価を行うための指標である．これらは，事前学習においてエンティティに関する事実を学習していると考えられることから，考案された指標となっている．実際に，破滅的忘却の抑制手法を言語モデルに適用することでこれらの指標の値が改善することが示されている．この 2 つの式については後の 3.2 節で述べる．

表 1 モデル設定

	Dense($E=1$)	MoE($E=8$)	MoE($E=64$)
active params	0.95B	0.34B	0.17B
Local Experts	1	8	64
Experts per token	1	2	2
FFN dim	65536	8192	1024

3 実験設定

本研究では，Liao らの先行研究 [3] で提案された方法に則って，Dense モデルと MoE モデルの事前学習時の破滅的忘却を評価・比較することとした．

3.1 学習・評価モデル

本研究では，一般公開されている GPT-OSS [12] を中間層の数やフィードフォワード（FFN）層の隠れ次元を変化させて，0.95B に調整した異なる 3 つのモデルを用いて比較実験を行った．具体的には，表 1 に示す 3 つの設定でモデルを構築した．ここで，E は Local Experts の数を表す．

ここで重要な点として，$E=1$ に設定した MoE モデルは Dense モデルと等価な設定となる．この関係により，MoE モデルでの Expert 数の設定の違いだけで Dense モデルと MoE モデルの比較評価を可能とした．

3.2 評価指標

(1) M_{in} **：** エンティティを含む文脈を言語モデルが入力として受け取ったときに，それに関連する情報を忠実に出力できるかを確かめるための指標である．具体的には，エンティティを終端に持つ 32 トークンをプロンプト $s_j \in S$ としてモデルに入力し，出力された 32 トークンを o_i $(i = 1, 2, \ldots, 32)$ とする．これに対して，オリジナルテキストにおいて，s_j に続く 32 トークンを t_i $(i = 1, 2, \ldots, 32)$ として，以下の式によって評価を行う．

$$M_{in} = \frac{\sum_{s_j \in S} \sum_{i=1}^{32} \mathbb{1}\{o_i = t_i\}}{32|S|} \tag{1}$$

(2) M_{ex} **：** エンティティの直前までの文脈を言

表 2 M_{in} の計算例

プロンプト	学習データ	モデル出力	M_{in}
日本 に ある 富士山	は 標高 3776 メートル	は 標高 3000 メートル	$\frac{1+1+0+1}{4} = 0.75$
日本 に ある 富士山	は 標高 3776 メートル	は 一番 高い 山	$\frac{1+0+0+0}{4} = 0.25$

表 3 M_{ex} の計算例

エンティティ	プロンプト	学習データ	モデル出力	M_{ex}
東京	日本 の 首都 は	東京 で あり，	東京 と いう 都市	1
東京	日本 の 首都 は	東京 で あり，	関東 に ある 東京	1
東京	日本 の 首都 は	東京 で あり，	京都 に 昔 は	0

語モデルが受け取ったときに，その後の 32 トークン以内にそのエンティティトークンを正しく出力できるかを測る指標である．具体的には，エンティティを含めずにその直前の 32 トークンをプロンプト $s_j \in S$ としてモデルに入力して，出力されたトークンを o_i $(i = 1, 2, \ldots, 32)$ とする．全評価サンプルに対して，この o_i にエンティティトークン c_j が含まれている割合を以下のような式で求めて，M_{ex} の値とする．

$$M_{ex} = \frac{\sum_{s_j \in S} \text{is\_substring}(c_j, \hat{o})}{|S|} \tag{2}$$

M_{in} と M_{ex} がどのように計算されるかの例を，それぞれ表 2，3 に示す．

3.3 評価データ

本研究では，学習データとして FineWeb-Edu [18] からサンプリングした計 25B トークンのコーパスを用いた[1)]．このコーパスをサブセット A（20B トークン）とサブセット B（5B トークン）に分割した．評価はエンティティ周辺のトークン列を対象に行うこととし，以下の手順で評価セットを構築した．

まず，English Wikipedia dataset[2)] から 400,000 件の記事タイトルをエンティティとして，無作為に抽出した．次に，これらの中から「サブセット A における出現頻度が上位 50%」かつ「サブセット B における出現頻度が下位 50%」という条件を満たすエンティティを集合 C を定義した．これは，サブセット A で一度学習した知識が，サブセット B の学習によってどの程度失われるかを測定するためである．サブセット A 内から集合 C に属するエンティティを抽出し，その前後 32 トークンを仮評価データとした．サブセット A の学習後，仮評価データで一度モデルの評価を行い，$M_{ex} = 1$ となるデータを，知識が定着しているデータとして，忘却量を測るための最終的な評価データとして採用した．今回は $E = 1, 8, 64$ のモデルに対して，それぞれ評価データは 1,897 件，1,971 件，1,845 件となった．

4 実験結果

3 章で示した評価方法に基づいて実験を行った結果のグラフを図 2 に示す．x 軸はサブセット A+B 全体の学習ステップ数を表しており，3.3 節で述べた通り，評価はサブセット B でのみ行ったため，グラフは 19,034 ステップ目から始まっている．

図 2 上側が表す M_{in} の評価結果においては，全てのモデルで学習を通して値が激しく振動する傾向が見られた．また，グラフの左端は，サブセット A の学習終了直後かつサブセット B の学習開始前の状態を表しているが，この時点でモデルに依らず一様に低い値を示している．これは，サブセット A に含まれるエンティティに続く文章の知識が定着していないことに加え，サブセット B の学習過程において，当該知識の想起と消失が繰り返していることを示唆している．以上の結果から，M_{in} では，学習中の忘却量を正確に測定できているとは言い難く，優劣の結論を導くことは困難である．

一方で，図 2 下側が表す M_{ex} の評価結果においては，3 つのモデルでサブセット B の学習初期段階において，値が 100%から 85%程へと減少することが観察された．また，Expert 数が多いほど，高い値を維持する傾向が見て取れる．この結果は，文脈に応じたエンティティの知識の保持において MoE モデルが Dense モデルよりも学習過程での忘却が生じにくい可能性があることを示唆している．

しかしながら，文献 [3] で報告された結果と比較した際に，これら 2 つのグラフの傾向が異なるため，MoE モデルの方が忘却抑制できていると断定はできず，今後より多角的な分析が必要となる．

1） https://huggingface.co/datasets/HuggingFaceFW/fineweb-edu
2） https://dumps.wikimedia.org/

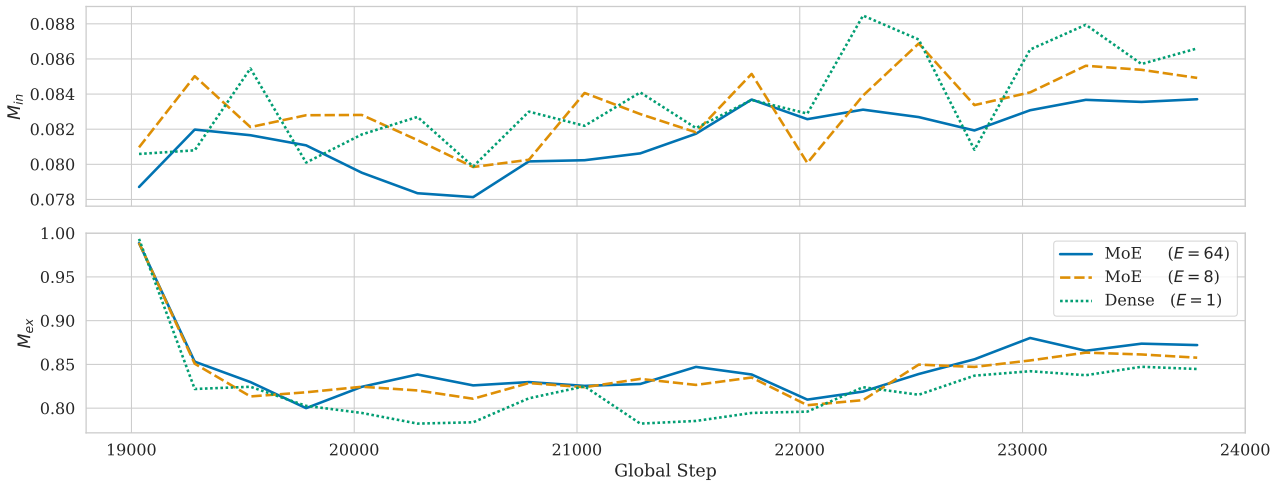

図 2　M_{in}，M_{ex} の評価結果．E はエキスパート数を表す．

5　考察

5.1　忘却に関する指標

図 2 を見ると，いずれの指標においても中盤辺りから僅かな増加傾向が確認された．この結果は，評価データに対する「忘却」が生じているのではなく，むしろ知識が回復している，あるいは汎化性能の向上が生じていることを示唆している．M_{in}，M_{ex} を提案した Liao らの先行研究 [3] においては，このような傾向は見られなかった．

文献 [3] においては，過学習を避けるために，1.5B のモデルサイズに対して，あえて少量である 13B トークンのサブセット A を用いていたのに対して，本実験では，モデルサイズに対するサブセット A の割合を大きくしたことが，異なる結果につながったではないかと考える．

本来であれば「忘却」という現象を他の学習・汎化プロセスから分離し，純粋に忘却量のみを定量化できる指標を用いることが望ましい．しかし，現状ではそのような指標は確立されておらず，設計には困難が伴うと予想される．今後，効率的な学習戦略を獲得するために極めて重要な検討課題と言える．

5.2　MoE による忘却抑制の要因

図 2 下側の M_{ex} の結果から，MoE モデルが Dense モデルに比べて学習中の忘却が抑えられる可能性があることが示唆された．これは，MoE において FFN 層がトークン毎にエキスパートを選択していることが大きな要因と言えると考えられる．一般的に複数の言語知識が FFN 層内の一つのパラメータに重ね合わされた状態で畳み込まれていると考えられている [19, 20]．破滅的忘却は，こうした重ね合わせの知識を無視して，一部の知識に応じた更新を行うために起こる現象であるとされている [1]．一方で，MoE アーキテクチャはトークン毎にエキスパートを選択するため，知識の重ね合わせが少なく，似た知識が同じエキスパートに分散して畳み込まれているとされている [14]．その上，パラメータ更新の際には，選択されたエキスパートのみに勾配が流れるため，現在の学習に関係のない知識に干渉する可能性が少なく，忘却が抑制されたのではないかと推察される．

このことを踏まえて今後，破滅的忘却を抑えながら効率的な学習を行うには，入力トークンに対する FFN 層の重みをニューロン単位で選択するなど，よりミクロな視点で知識を分散させるような仕組みや，パラメータを選択的に更新するような仕組みが重要になってくるのではないかと考えられる．これにより，パラメータ上での知識の干渉をより少なくし，忘却の抑制が見込めるのではないかと予想される．

6　おわりに

今回は，言語モデルにおける MoE の構造的特性に着目し，破滅的忘却が少ないであろうという仮説を立てて定量評価を行った．その結果，MoE モデルの方が Dense モデルに比べて，学習を通して忘却量が少ない可能性を示唆する結果が得られた．しかし，文献 [3] で報告された結果と異なる点が多いため，仮定が正しいと断定付けることは難しく，今後の多角的な分析が必要とされる．

謝辞

本研究は，JST ムーンショット型研究開発事業 JPMJMS2011-35 (fundamental research)，および，文部科学省の補助事業「生成 AI モデルの透明性・信頼性の確保に向けた研究開発拠点形成」の助成を受けたものです．また，本研究成果の一部は，北海道大学情報基盤センターのスーパーコンピュータ Grand Chariot 2，および，「ABCI 3.0 開発加速利用」の支援を受けて産総研及び AIST Solutions が提供する ABCI 3.0 を利用して得られたものです．

参考文献

[1] R M French. Catastrophic forgetting in connectionist networks. **Trends Cogn. Sci.**, Vol. 3, No. 4, pp. 128–135, April 1999.

[2] Tongtong Wu, Linhao Luo, Yuan-Fang Li, Shirui Pan, Thuy-Trang Vu, and Gholamreza Haffari. Continual learning for large language models: A survey. **arXiv [cs.CL]**, February 2024.

[3] Chonghua Liao, Ruobing Xie, Xingwu Sun, Haowen Sun, and Zhanhui Kang. Exploring forgetting in large language model pre-training. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, **Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 2112–2127, Vienna, Austria, July 2025. Association for Computational Linguistics.

[4] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. **arXiv [cs.LG]**, January 2020.

[5] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. **arXiv [cs.CL]**, March 2022.

[6] Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. Position: Will we run out of data? Limits of LLM scaling based on human-generated data. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, **Proceedings of the 41st International Conference on Machine Learning**, Vol. 235 of **Proceedings of Machine Learning Research**, pp. 49523–49544. PMLR, 21–27 Jul 2024.

[7] Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In **International Conference on Learning Representations**, 2017.

[8] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. GShard: Scaling giant models with conditional computation and automatic sharding. In **International Conference on Learning Representations**, 2021.

[9] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. **arXiv [cs.LG]**, January 2021.

[10] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts. **arXiv [cs.LG]**, January 2024.

[11] DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, et al. DeepSeek-V3 technical report. **arXiv [cs.CL]**, December 2024.

[12] OpenAI, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, et al. gpt-oss-120b & gpt-oss-20b model card. **arXiv [cs.CL]**, August 2025.

[13] Weilin Cai, Juyong Jiang, Fan Wang, Jing Tang, Sunghun Kim, and Jiayi Huang. A survey on mixture of experts in large language models. **IEEE Trans. Knowl. Data Eng.**, Vol. 37, No. 7, pp. 1–20, 2025.

[14] Hongbo Li, Sen Lin, Lingjie Duan, Yingbin Liang, and Ness Shroff. Theory on mixture-of-experts in continual learning. In **The Thirteenth International Conference on Learning Representations**, 2025.

[15] Wuyang Chen, Yanqi Zhou, Nan Du, Yanping Huang, James Laudon, Zhifeng Chen, and Claire Cui. Lifelong language pretraining with distribution-specialized experts. In **Proceedings of the 40th International Conference on Machine Learning**, ICML'23. JMLR.org, 2023.

[16] Grzegorz Rypeść, Sebastian Cygert, Valeriya Khan, Tomasz Trzcinski, Bartosz Michał Zieliński, and Bartłomiej Twardowski. Divide and not forget: Ensemble of selectively trained experts in continual learning. In **The Twelfth International Conference on Learning Representations**, October 2023.

[17] Soochan Lee, Junsoo Ha, Dongsu Zhang, and Gunhee Kim. A neural dirichlet process mixture model for task-free continual learning. In **International Conference on Learning Representations**, September 2019.

[18] Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, **Advances in Neural Information Processing Systems**, Vol. 37, pp. 30811–30849. Curran Associates, Inc., 2024.

[19] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. **arXiv [cs.LG]**, September 2022.

[20] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing**, pp. 5484–5495, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

A　付録

A.1　ハイパーパラメータ

今回の学習で用いたハイパーパラメータは，以下の表 4 に示した通りである．

表 4　ハイパーパラメータ

パラメータ名	値
Global Batch Size	1024
Sequence Length	1024
Optimizer	AdamW
Adam beta1	0.9
Adam beta2	0.95
max LR	3e-4
min LR	3e-5
LR Scheduler Type	WSD
Warmup Steps	625
Decay Steps	2384
Decay type	cosine

A.2　実験の詳細

今回の学習において得られた Loss のグラフは以下の図 3 のようになった．また，Learning Rate のスケジューラは図 4 に示したように機能した．ここで，19,034 ステップ目で LR が急激に下がっている部分があるが，これは WSD Scheduler によるもので，サブセット B の学習に切り替わったときに 1 ステップだけ表 4 に示した min LR の値で更新されたことを表している．

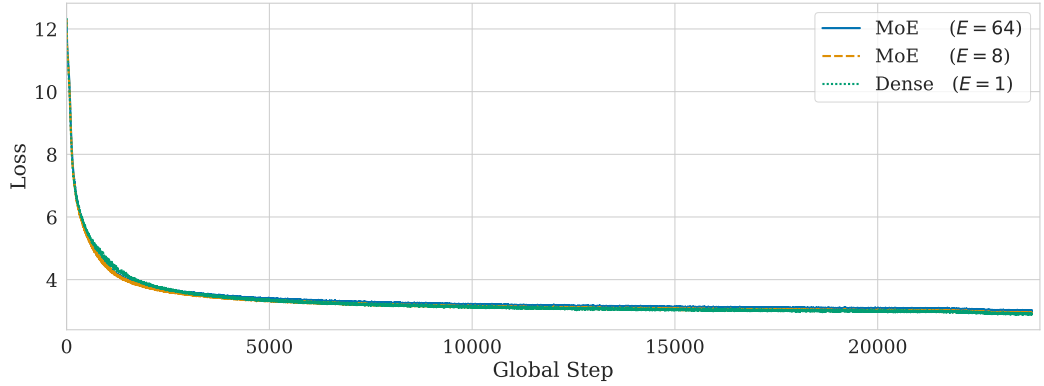

図 3　Loss の曲線．E はエキスパート数を表す．

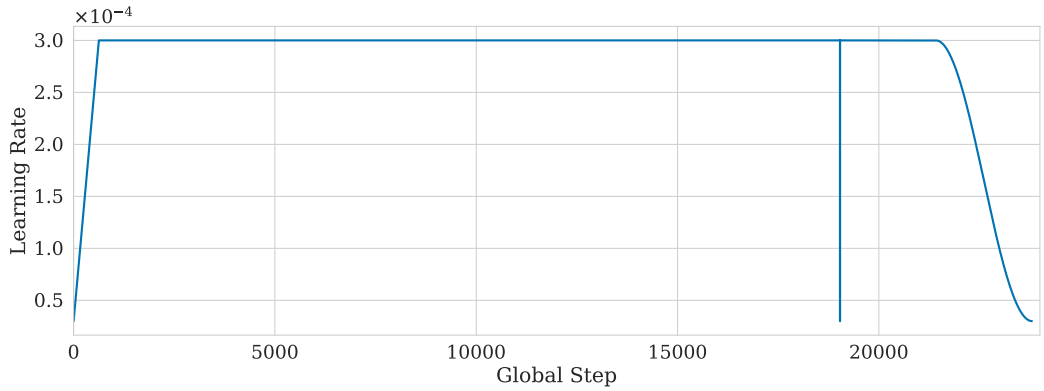

図 4　Learning Rate